

# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ  
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY  
DEPARTMENT OF INFORMATION SYSTEMS

## INFORMAČNÍ SYSTÉM PRO DOMÁCNOST

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

MICHAIL KNYREVICH

BRNO 2009



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ  
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ  
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY  
DEPARTMENT OF INFORMATION SYSTEMS

# INFORMAČNÍ SYSTÉM PRO DOMÁCNOST

HOME INFORMATION SYSTEM FOR

BAKALÁŘSKÁ PRÁCE  
BACHELOR'S THESIS

AUTOR PRÁCE  
AUTHOR

MICHAIL KNYREVICH

VEDOUCÍ PRÁCE  
SUPERVISOR

ING. IGOR SZÖKE

BRNO 2009

## **Abstrakt**

Tato bakalářská práce se zabývá tvorbou Informačního systému pro domácnost a to především jeho modulem Energie. Jsou zde analyzované již existující informační systémy zabývající se tímto tématem. Dále je zde popsán formální návrh vlastního modulu Energie na základě získaných informací. Součástí tohoto návrhu je systém predikce výdajů za energie, který se provádí na základě monitorování počasí. Na závěr je realizace a testování tohoto návrhu s použitím webových technologií a to především Zend Framework.

## **Klíčová slova**

Informační systém pro domácnost, Energie, Zend Framework, MVC, Webové technologie, Lineární regrese

## **Abstract**

This bachelor's thesis involves the development of the Informational system for home, especially its Energy module. Some existing informational systems of this topic are analyzed there. Further, there is a formal concept of self Energy module, which is based on gained information. The part of this concept is prediction system of energetic expenses, which uses a weather monitoring. Realization and testing with usage of web technologies (especially the Zend Framework) are placed at the end of my thesis.

## **Keywords**

Information system for home, Energy, Zend Framework, MVC, Web technologies, Linear regression

## **Citace**

Knyrevich Michail: Informační systém pro domácnost, bakalářská práce, Brno, FIT VUT v Brně, 2009

# Název bakalářské práce v jazyce práce

## Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením Ing. Igora Szöke

Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....  
Knyrevich Michail  
20.5.2009

## Poděkování

Rad bych poděkoval svému vedoucímu Ing. Igoru Szökemu za jeho čas, cenné rady a informace.

Také chci poděkovat svojí rodině a přítelkyni za podporu a gramatickou kontrolu textu.

© Knyrevich Michail, 2009.

*Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů.*

# Obsah

Obsah.....	1
1 Úvod.....	3
2 Informační systém pro domácnost.....	4
2.1 Obecné informace.....	4
2.1.1 Modul Finance.....	4
2.1.2 Modul Vozidla.....	5
2.1.3 Modul Energie.....	5
2.1.4 Modul Stravování.....	5
2.1.5 Modul Bydlení.....	5
2.2 Výdaje za energii v domácnosti.....	5
2.2.1 Elektrická energie.....	6
2.2.2 Zemní plyn.....	6
2.2.3 Voda.....	6
2.2.4 Alternativní zdroje.....	6
2.3 Programy pro práci s výdaji za energie.....	7
2.3.1 Energie pod palcem 1.09.....	7
2.3.2 Platby elektrické energie 1.0.0.0.....	7
2.3.3 Domácí hospodářství 11.0.....	7
2.3.4 Vyhodnocení průzkumu.....	8
2.4 Práce s energiemi v IS.....	8
3 Návrh.....	9
3.1 Formální návrh.....	9
3.1.1 Základní vlastnosti systému.....	9
3.1.2 Funkce systému.....	9
3.2 Formální návrh - tabulky a formuláře.....	10
3.2.1 Energie.....	10
3.2.2 Měřidla.....	10
3.2.3 Intervaly .....	10
3.2.4 Přehledy.....	10
3.2.5 Predikce výdajů.....	11
3.3 E-R diagram.....	11
3.3.1 Popis E-R diagramu.....	13
3.4 Návrh grafických prvků.....	13

4 Implementace.....	14
4.1 Zend Framework.....	14
4.1.1 MVC (Model-View-Controller).....	14
4.1.2 Stromová struktura.....	15
4.2 Implementace prostředí.....	15
4.2.1 Práce s daty.....	16
4.3 Predikce výdajů.....	17
4.3.1 Lineární regrese.....	17
4.3.2 Lineární regrese - rovnice.....	17
4.4 Grafické prvky.....	18
5 Testování.....	19
6 Další možný vývoj.....	20
7 Závěr.....	21
Literatura.....	22
Seznam příloh.....	23
Příloha 1.....	24

# 1 Úvod

Cílem této práce bylo navrhnout a implementovat Informační systém pro domácnost, a především jeho modul Energie. Tento systém by měl zajistit větší přehled ve výdajích za energii v domácnostech. Dále by měl program odhadovat výdaje na základě počasí a výdajů z předchozích let. Systém by měl splňovat nároky kvalitního softwaru: bezchybná funkčnost, nenáročnost a uživatelsky přívětivé prostředí. Hlavní součástí práce by měl být kvalitní návrh, jehož lze použít nejen pro tento Informační systém, ale i pro návrh jiných podobných programů a aplikací.

Úvod této práce je věnován seznámení s pojmem Informační systém pro domácnost a využití těchto typů systému. Dále budou uvedeny mé poznatky v oblasti výdajů domácností za energie. V této části budou zmíněné způsoby výpočtu plateb, paušálu a záloh za různé druhy energií. V dalších kapitolách bude podrobně popsán formální návrh informačního systému. Tato část nám nejvíce přiblíží všechny procesy, které by měly v aplikaci tohoto typu probíhat. Bude zde popsána datová část, návrh některých důležitých systémů a jejich vzájemné propojení. Předposlední kapitola bude věnována samotné implementaci programu. Také bych zde rád popsal prostředí Zend Frameworku, které jsem si vybral pro vývoj tohoto systému. Dále v této kapitole najdete celkový popis vytváření aplikace, od prvních návrhů formulářů až po systém predikcí. Při popisu všech funkcí systému budu klást důraz na popis zajímavých programovacích postupů. Poslední kapitoly se budou věnovat testování systému a celkovému zhodnocení práce, kde bych chtěl naznačit další vývoj systému a poukázat na některé nedostatky a také přínosy celého programu.

## 2 Informační systém pro domácnost

V této kapitole jsou uvedeny základní vlastnosti, kterými by měl informační systém pro domácnost disponovat. Podíváme se také, jak by se v takovýchto systémech mělo pracovat s energiemi. Na konci bude uveden přehled již existujících programů, zabývajících se touto tematikou.

### 2.1 Obecné informace

Informační systémy pro domácnost se od obyčejných informačních systémů liší hlavně v počtu lidí, které s nimi pracují. Na rozdíl od podnikových informačních systémů, jsou obsluhovány pouze jedním až deseti uživateli. Malý počet uživatelů přináší několik specifických požadavků, které se liší od zaběhnutých procesů vytváření informačních systémů:

- není potřeba vytvářet účty (správa uživatelů).
- větší nastavitelnost systému, menší standardizace.

Uživatelské účty nejsou potřebné hlavně díky malému počtu uživatelů. Také je většinou v domácnosti pouze jedna osoba, která se stará o chod financí a tím pádem pracuje s informačním systémem. Nastavitelnost systému je důležitá, jelikož každá domácnost je jiná a využívá jiné zdroje. Nemůžeme mít např. systém pouze pro jeden druh energie. Spousta domácností využívá kombinaci nejrozumnějších energetických zdrojů a tato fakta musí být v systému zohledněna.

Informační systém pro domácnost by měl obsahovat následující:

#### 2.1.1 Modul Finance

Většinou nejdůležitější modul celého systému. Informační systém pro domácnost by měl hlavně usnadnit a zpřehlednit pohyby financí v domácnosti. Proto většina vstupů v systému bude směřovat právě do modulu Finance. Modul by měl poskytovat přehled výdajů a příjmů, které by měly být rozděleny na hotovostní a bezhotovostní položky. Jeden velký oddíl by měl zahrnovat správu bankovníctví a bankovních účtů v rodině. Samozřejmostí by měly být přehledy výdajů za určité období (měsíční, kvartální, roční).



### **2.1.2 Modul Vozidla**

Tento modul by měl umět pracovat s více vozidly. Pro každé z vozidel by se měla vést kniha jízd a spotřeba pohonných hmot. Systém by měl evidovat technické kontroly, které vozidlo absolvovalo a upozorňovat na nutnost nové prohlídky. Pro každé vozidlo by měly existovat záznamy o opravách a příslušných nákladech.

### **2.1.3 Modul Energie**

Podstatnou funkcí tohoto modulu je kontrola výdajů za energie a vypočítávání vrácení záloh. Dále by měl umožnit pracovat s více druhy energie, jelikož většina domácností je kombinuje. Systém by měl pracovat s měřidly a zaznamenávat jejich stavy. Volitelná je pro tento modul je správa elektrických spotřebičů, které také mohla být v modulu Bydlení viz. kapitola 2.1.5. Podrobnějšímu popisu tohoto modulu je věnována samostatná kapitola 2.4.

### **2.1.4 Modul Stravování**

V tomto modulu se shromažďují informace o výdajích za potraviny. Modul by měl umět vytvářet nákupní seznamy, hlásit nedostatek konkrétních surovin a vést přehled a historii cen. Také by mohl zahrnovat i ostatní výdaje za jídlo jako jsou například návštěvy různých stravovacích zařízení.

### **2.1.5 Modul Bydlení**

Tento modul se zabývá náklady na bydlení. Měl by umět pracovat s více nemovitostmi. Evidovány by měly být veškeré náklady na opravy a inovace, stejně jako na nábytek. Součástí by měly být tzv. budíky, které by připomínaly různé kontroly a další naplánované události (služby na chodbách, kontrola plynu, kontrola jističů atd.).

## **2.2 Výdaje za energii v domácnosti**

Jenom malé procento domácností využívá pouze jeden zdroj energie. Z ekonomického hlediska je výhodnější je kombinovat. Zde jsou uvedeny nejrozšířenější zdroje energie a náklady s nimi spojené:

### 2.2.1 Elektrická energie

Nejrozšířenější zdroj energie v domácnostech. Roční náklady s ním spojené:

- platby za spotřebu VT (vysoký tarif) = počet MWh \* tarif VT;
- platby za spotřebu NT (nízký tarif) = počet MWh \* tarif NT;
- stálé platby = 12 \* (tarif podle typu jističe \* pevná cena za měsíc podle tarifu).

Pro menší zatěžování rodinného rozpočtu jsou tyto platby přepočítány na měsíční platby podle hodnoty jističe a předchozí spotřeby domácnosti. Poté jsou vždy jednou za rok vypočítány rozdíly mezi odhadovanou a reálnou spotřebou, které jsou pak vyrovnané. [1]

### 2.2.2 Zemní plyn

U zemního plynu jsou roční náklady podobné jako u elektrické energie:

- platba za spotřebu plynu = cena kWh \* suma odebraných kWh;
- stálé platby = 12 \* stálý měsíční plat podle přidělené roční spotřeby.

Dříve se spotřeba plynu počítala v m<sup>3</sup>, ale od roku 2001 se používá jednotka kWh, přičemž vzorec pro převod je: 1 m<sup>3</sup> = 10,55 kWh. [2]

### 2.2.3 Voda

Platba za vodu se počítá podle vzorce:

$$\text{spotřebovaná voda v m}^3 * (\text{cena vodného za m}^3 + \text{cena stočného za m}^3)$$

Možnost jak snížit tyto náklady, jsou vlastní prostředky pro odtok vody. Jedná se o využití vody na venkovních pozemcích nebo zahradách. V tomto případě je výhodné umístit speciální měřidlo jen pro tento odběr vody. Na konci roku je pak možné odečíst od celkových nákladů stočné za m<sup>3</sup> z venkovního měřidla.

### 2.2.4 Alternativní zdroje

Do této kategorie spadají méně rozšířené zdroje energie, např. solární energie, bio energie, větrná energie atd. Tyto zdroje většinou slouží jako podpůrné a úsporné energie, kde největší náklady vycházejí z pořízení zpracovávajícího zařízení.

## **2.3 Programy pro práci s výdaji za energie**

Důležitým krokem pro vytvoření informačního systému je průzkum již existujících softwarových výrobků s tímto zaměřením. Po prozkoumání internetu jsem několik takových programů našel a otestoval.

### **2.3.1 Energie pod palcem 1.09**

Jednoduchý program pro zadávání a kontrolu výdajů za energie. Aplikace funguje pod platformou Windows a pro svůj chod nepotřebuje žádný další software. Podporuje export a import údajů. Dokáže sledovat ceny přes internet. Program pracuje s obdobími a produkuje grafické a textové statistiky. Velice zajímavou funkcí je "Hlídač", který vám vždy na konci měsíce připomene zaplacení paušálů.

Nevýhody: pouze pro Windows, jen celkové výdaje - nelze určit více spravovaných objektů, Shareware. [3]

### **2.3.2 Platby elektrické energie 1.0.0.0**

Kalkulačka ročních výdajů za elektřinu. Pracuje na platformě .NET. Po zadání potřebných údajů jako jsou například cena za MWh a cena systémových služeb atd. lze vypočítat přesnou částku, kterou za rok zaplatíme. [4]

Nevýhody: instalace .NET, v podstatě pouze jediná funkce, stále se jedná o beta verzi.

### **2.3.3 Domácí hospodářství 11.0**

Rozsáhlý program pro správu domácnosti. Zaměřil jsem se na prozkoumání práce s výdaji energie. Program pracuje se třemi základními druhy energie: elektrická energie, plyn, voda. Počítá s denním a nočním proudem. Určuje, kolik peněz se vrátí ze záloh. Vše se zařazuje do celkových výdajů a příjmů. Celkově program působí komplexně a je podle mě nejlepší volbou pro kontrolu energetických výdajů domácnosti. [5]

Nevýhody: Shareware, mírně složitější ovládání, omezení pouze na 3 typy energie.

### **2.3.4 Vyhodnocení průzkumu**

Průzkum ukázal, že nástroje na správu energetických výdajů v domácnosti existují, avšak obsahují určité nedostatky a programy s větší funkcí nejsou dostupné jako Freeware. Nevýhodou je také kompatibilita těchto aplikací výhradně s OS Windows. Dále jsem se dozvěděl o zajímavých funkcích, které by můj Informační systém mohl obsahovat, a které jsou v tomto kontextu velice užitečné. Jedná se o tzv. Hlídače, který nás upozorňuje na zaplacení paušálu a vzorec pro výpočet a prezentování výsledných částek, které za elektřinu zaplatíme.

## **2.4 Práce s energiemi v IS**

Základními požadavky na obecný IS pro domácnost jsou:

- schopnost pracovat se základními druhy energie;
- pracovat paušálními výdaji energie;
- spravovat měřidla;
- umět nastavovat různé ceny energií pro různá období;
- pracovat se zálohami na energii a provádět výpočty skutečných výdajů za energii;
- vytvářet přehledy o výdajích, nejlépe roční.

## 3 Návrh

Po prostudování všech podkladů se dostáváme k fázi návrhu. Kvalitní návrh je základem každého IS a proto jsem pro svůj projekt využil dva modelové prostředky.

- **Formální slovní návrh** - definice tabulek, formulářů a funkčnosti systému
- **ER diagram** (Entity-relationship diagram)

V úvahu ještě přicházel *Diagram případů užití*, ale ten jsem nakonec vynechal, jelikož aplikace byla vymyšlena jako jednouživatelská.

### 3.1 Formální návrh

V této části je slovně popsán návrh funkčních prvků systému a návrh jednotlivých podsekcí modulu Energie.

#### 3.1.1 Základní vlastnosti systému

Nejdříve bylo nutné se rozhodnout, jak samotný systém realizovat. V úvahu přicházela buď desktopová, nebo webová realizace. V tomto kontextu jsem zvážil několik kritérií:

- multiplatformní aplikace;
- přístupnost z více počítačů (domov, práce);
- jednoduchá instalace.

Nakonec jsem zvolil vytvoření IS pomocí webových technologií, jelikož splňovaly výše uvedená kritéria. Dalším důvodem byla absence větších zkušeností s webovými technologiemi, které bych si na tomto projektu chtěl vyzkoušet.

#### 3.1.2 Funkce systému

Celý informační systém by měl být modulární, tj. systém by se měl dělit na základ aplikace a několik dalších logických částí. Toto pojetí zajistí přehlednost zobrazovaných dat a jednodušší orientaci v nabídkách aplikace.

Každý systém by měl splňovat určité požadavky na zobrazení dat. Základním kritériem je, aby se data zobrazovala přehledně, a aby vazby mezi nimi byly na první pohled zřetelné. Tuto úlohu splňuje tabulkové zobrazení, které bude tento IS podporovat. Pro korektní zobrazení dat v tabulkách je zapotřebí ještě několik funkcí:

- Filtrace sloupců;
- Setřídění dat;
- Editace záznamů v novém okně.

Funkce editace záznamu může probíhat buď *editací v novém okně*, nebo *editací přímo v tabulce*.

Z uvedených způsobů jsem zvolil první variantu, a to kvůli jednodušší implementaci a snadnějšímu ovládání.

Přesto že je systém navrhnut jako jednouživatelská aplikace, bude při jeho spuštění vyžadována autentizace uživatele pomocí přihlašovacího jména a hesla. Je to kvůli možnému umístění aplikace na webu, kde by k datům mohli přistupovat i jiní uživatelé a mohli by data zneužít.

## 3.2 Formální návrh - tabulky a formuláře

Základem každého informačního systému jsou tabulky a formuláře. Náš modul Energie bude obsahovat několik kategorií:

### 3.2.1 Energie

Tato sekce modulu bude obsahovat tabulky pracující s energiemi, paušály za energie, jednotky energie, a ceny za jednotlivé energie. Také bude obsahovat aktuální přehled výdajů za ně. Tento přístup umožní uživateli vytvářet si a nastavovat energie podle své potřeby.

### 3.2.2 Měřidla

Zde se budou vyskytovat tabulky a formuláře spravující měřidla, jejich stavy, období a data resetování měřidel, zúčtovací období měřidel. Stavy jednotlivých měřidel se budou zadávat uživatelem ručně a to v náhodných intervalech.

### 3.2.3 Intervaly

V této sekci budou tabulky pracující s jednotlivými obdobími, které budou určovat dobu trvání cen jednotek, paušálu atd. Období se budou dělit na předdefinované (měsíce, roky) a uživatelsky definované.

### 3.2.4 Přehledy

Sekce bude obsahovat různé přehledy a porovnání výdajů:

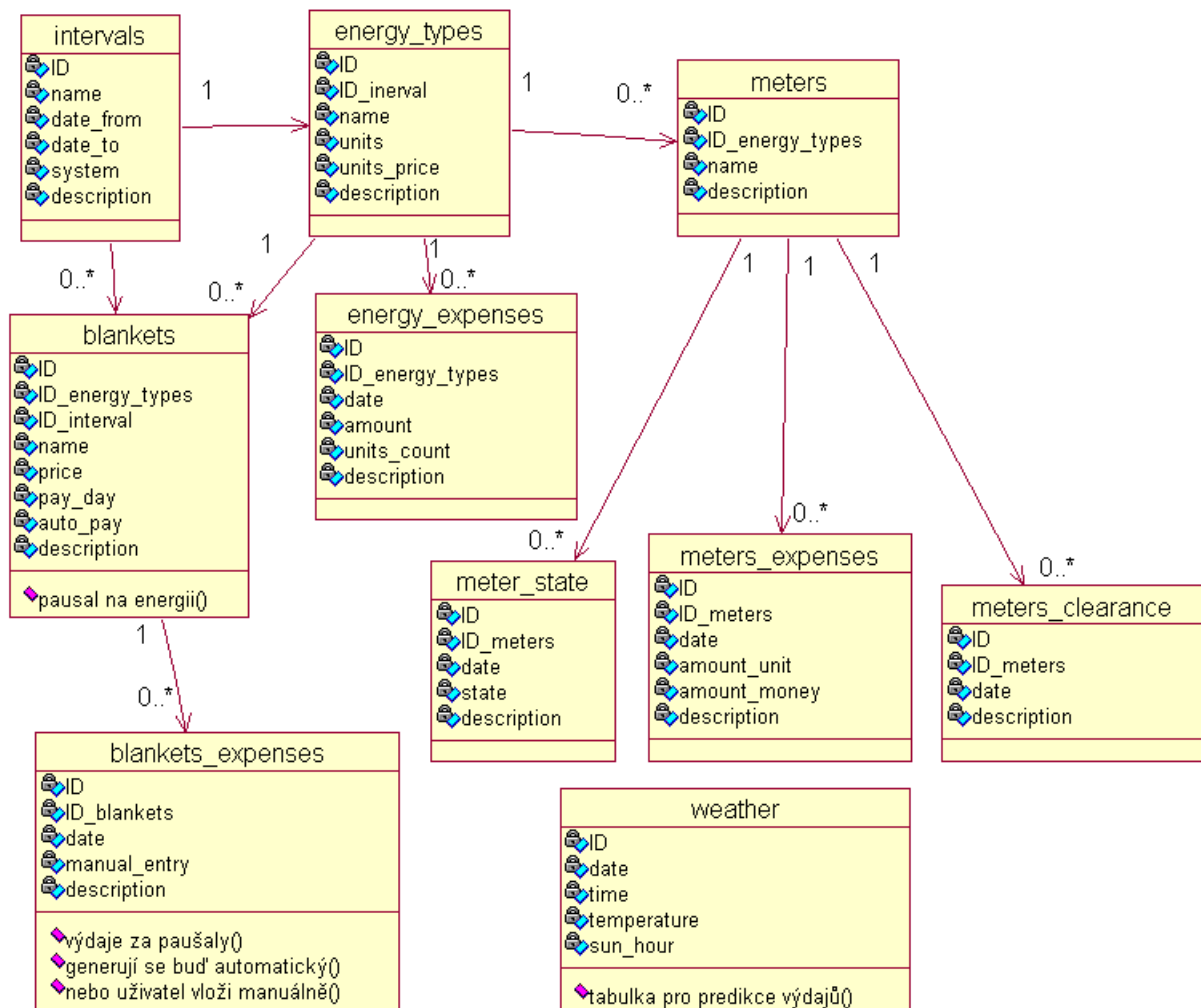
- Výpočet doplatku nebo přeplatku za energii;
- Roční výdaje za energii.

### 3.2.5 Predikce výdajů

Tato část našeho modulu bude provádět predikci výdajů na základě venkovních teplot a výdajů za energii z minulých let. Pro predikci bude tato funkce využívat Lineární regresi. Uživatel si bude moci zvolit, ze kterého období použít vzorek dat, pro jaké období budeme chtít předpovědět výdajem, a o kolik se má změnit průměrná teplota v tomto období.

## 3.3 E-R diagram

E-R diagram slouží k modelování dat a jejich vztahů v klidovém stavu. Používáme jej při návrhu Databáze a při prezentaci návrhu zákazníkům. Z ER diagramu lze přechíst, které tabulky se musí vyskytnout v aplikaci a jak jsou mezi sebou propojené. Na diagramu můžeme najít entity, které reprezentují tabulky, vztahy mezi entitami a atributy entit.[6]



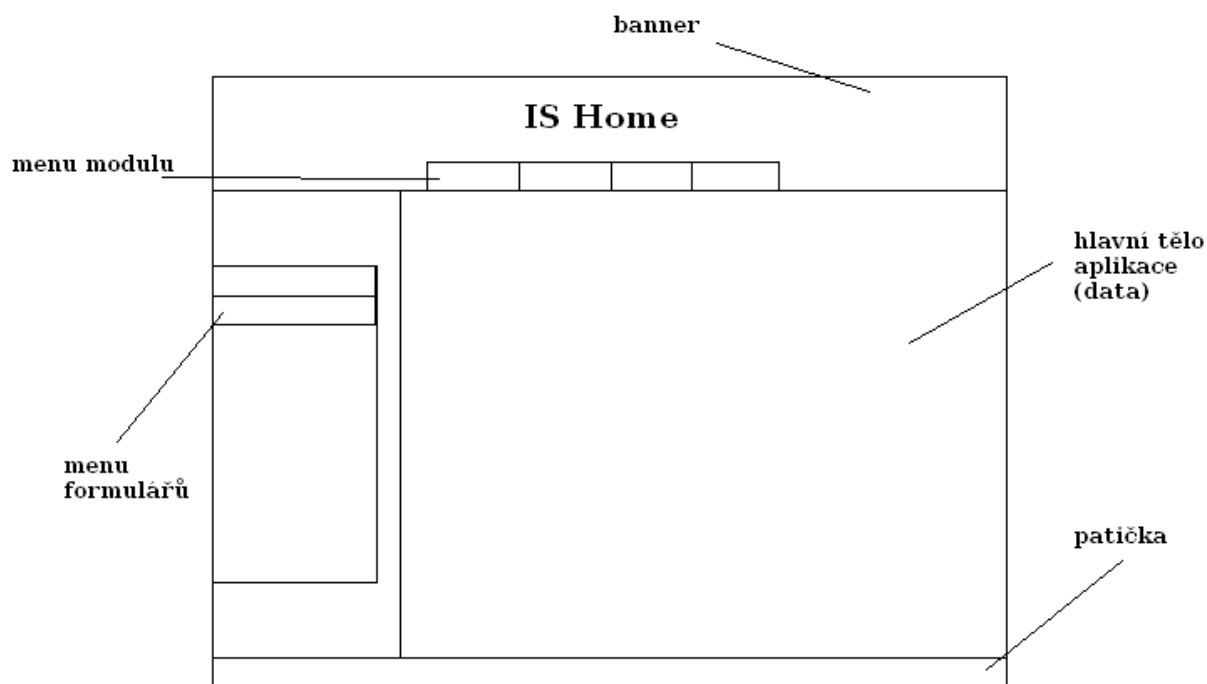
### 3.3.1 Popis E-R diagramu

Většina entit vychází z entity **energy\_types**, která určuje jednotky a ceny energií. Použití tohoto konceptu umožňuje uživateli vytvářet vlastní typy energií. Pro každou z těchto energií může vytvořit více paušálů, např. pro Nízký tarif a Vysoký tarif u elektrické energie. Entita **intervals** umožňuje definici běžných i více specifitějších časových intervalů, např. rok 2009, květen 2009, 1. čtvrtletí 2009 atd. Toto řešení nejenom šetří místo v databázi, ale také umožňuje větší přehled nad intervaly a následně jednodušší výpočet výdajů.

Entita **weather** se používá pro ukládání stavu počasí, pro jednotlivé dny. Tyto informace se poté používají při predikci výdajů. Proto entita není ve vztahu s ostatními entitami.

## 3.4 Návrh grafických prvků

V rámci návrhu jsem také provedl rozložení jednotlivých grafických a ovládacích prvků systému, který zahrnuje menu modulů, menu formulářů a hlavní část aplikace, kde se budou zobrazovat data. Zde je původní návrh:



V rámci grafického návrhu jsem se také rozhodl, že jednotlivé moduly budou využívat různé palety barev pro snazší orientaci.



## 4 Implementace

Tato kapitola se zabývá samotným vývojem Informačního systému. Popisuje jednotlivé kroky při implementaci všech důležitých částí programu.

### 4.1 Zend Framework

Pro implementaci Informačního systému je většinou vhodné vybrat vývojové prostředí (framework), který ulehčí samotný vývoj. Takových vývojových prostředí existuje hned několik, ale po prostudování dostupných materiálů [7] jsem se rozhodl pro Zend Framework.

Tento framework se využívá pro tvorbu webu a podporuje spoustu moderních webových technologií jako jsou (PHP, XHTML, CSS, jQuery, AJAX, a další, více informací zde [8]). Primárně je však orientován na skriptovací jazyk PHP. Zend je modulárně orientovaný a díky této vlastnosti můžeme s naší aplikací distribuovat pouze ty moduly, které potřebujeme.

#### 4.1.1 MVC (Model-View-Controller)

Pod touto zkratkou se skrývá podstata práce se Zend Frameworkem. Jedná se o třívrstvou architekturu, kde jsou jednotlivé kroky zpracování dat od sebe odděleny.

<b>Model</b>	Model zaštiťuje samotná data, se kterými aplikace pracuje. Vytváří nad nimi abstraktní vrstvu, se kterou pak lze velice pohodlně pracovat. Mohou to být data z databáze nebo jiného uspořádaného souboru dat.
<b>View (Zobrazení)</b>	Tato vrstva se stará o zobrazení zpracovávaných dat uživateli. Pro tyto účely využívá především XHTML, ale i další technologie.
<b>Controller (Ovládač)</b>	Controller provádí propojení Model a View vrstev. Nejdříve model vytvoří, a poté mu přiřadí správný pohled. Mezi tím také může samotná data upravovat.

Tato vlastnost umožňuje větší přehled nad vytvářenou aplikací a zajišťuje jednodušší údržbu.

## 4.1.2 Stromová struktura

Pro správný chod aplikace je zapotřebí dodržovat předepsanou souborovou strukturu.

```
application/  
  controllers/  
  models/  
  views/  
    filters/  
    helpers/  
    scripts/  
library/  
tests/  
webroot/
```

Složka **application** obsahuje tři základní části aplikace: **controllers**, **models** a **views**. Poslední se dělí ještě na tři podsložky:

- **filters** - obsahuje pomocné soubory pro filtraci zobrazených dat
- **helpers** - obsahuje pomocné PHP skripty pro co nejlepší kontrolu nad **views**. Hlavním přínosem je opětovná použitelnost často se vyskytujícího kódu.
- **scripts** - složka obsahuje samotné pohledy, které jsou většinou ve formátu \*.phtml.

V složce **library** se nachází externí PHP knihovny. Jsou zde umístěny i samotné knihovny Zend Frameworku. Složka **webroot** obsahuje obrázky, kaskádové styly (CSS) a soubory JavaScript. Hlavní soubor **index.php** je umístěn buď v kořenovém adresáři aplikace, nebo ve složce **webroot**. [9]

## 4.2 Implementace prostředí

Systém se skládá z několika tabulek a formulářů. Zpracování těchto částí systému mají na starosti jednotlivé controllery a views. Během práce jsem dospěl k ucelené šabloně pro jejich tvorbu. Zde si popíšeme, jak taková šablona vypadá.

Controller se většinou skládá z pěti funkcí:

- **init()**
  - Tato funkce se provede před každým Action v controlleru. Načítají se zde databázové modely ošetření pro menu formulářů, popisky které se využijí ve všech funkcích a další nastavení.
- **indexAction()**
  - Zde se provede načtení dat z databáze, které se odešlou na zobrazení.
- **addAction()**
  - V této funkci se nejdříve vytvoří formulář. Když je formulář správně vyplněn a odeslán, data se uloží do databáze.
- **editAction()**

- Editace dat v databázi. Velice podobný kód jako v `addAction()`. Při vytvoření se formulář navíc naplní daty.
- **deleteAction()**
  - Funkce smazání záznamu. Každé vymazání je potřeba potvrdit ve speciálním okně.

Slovo **Action** v názvu říká, že tuto funkci můžeme spustit v rámci prohlížeče. Provedeme to tak, že název funkce napíšeme s lomítkem za název controlleru (**localhost/ISHome/energie/add**). Pro každý **Action** musí existovat zvláštní view ve složce scripts, např. **edit.phtml**, který se stará o jeho zobrazení uživateli.

## 4.2.1 Práce s daty

Pro jednodušší zadávání intervalu a jiných položek s daty jsem použil volně šiřitelnou knihovnu **JS Calendar**. [10]

?	May, 2009							×
«	<	Today					>	»
wk	Sun	Mon	Tue	Wed	Thu	Fri	Sat	
17						1	2	
18	3	4	5	6	7	8	9	
19	10	11	12	13	14	15	16	
20	17	18	19	20	21	22	23	
21	24	25	26	27	28	29	30	
22	31							
Time:			14	:	50			
Mon, May 25 (today)								

Pokud víme, že budeme v našem formuláři používat pole pracující s daty, musíme v controlleru nastavit proměnnou **useDate** na **true**. Dalším krokem je přidání souboru ze složky **MyDate**, na konec zobrazení. Tyto soubory provedou inicializaci okna kalendáře a určí prvek, kam se bude vybrané datum ukládat.

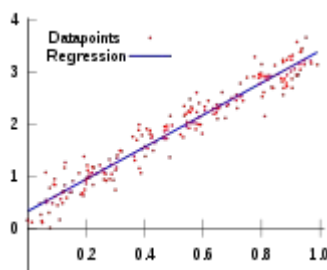
## 4.3 Predikce výdajů

Tato funkce systému dokáže předpovědět výdaje za energii na základě předchozích výdajů a stavu počasí. Algoritmus predikce je založen na Lineární regresi (závislosti) těchto veličin.

### 4.3.1 Lineární regrese

Lineární regrese je zkoumání závislosti mezi dvěma či více veličinami. Pro tyto účely je potřeba mít určitý vzorek dat, u kterých zkoumáme jejich vzájemnou závislost. Existují dva různé pohledy na lineární regresi:

**Lineární regrese pomocí aproximací křivky** - tento princip se používá u závislosti dvou veličin. V zkratce jde o nalezení takové rovnice přímky, abychom jí mohli proložit množinu bodů v grafu s co nejmenší odchylkou.



**Obecná lineární regrese** - tato metoda využívá k nalezení závislosti lineární kombinaci jakýchkoliv funkcí. Do této oblasti spadají nejen aproximace pomocí křivky, ale také pomocí polynomu nebo paraboly.

Pro naše účely bude postačovat lineární regrese pomocí aproximací křivky.

### 4.3.2 Lineární regrese - rovnice

Základním vzorcem v našem případě je obecná rovnice přímky:

$$Y' = b_{yx} * x + a_{yx}$$

- $b_{yx}$  je regresní koeficient,
- $a_{yx}$  je absolutní člen,
- $x$  je parametr rovnice (v našem případě to bude průměrná teplota)
- $Y'$  je teoretický odhad funkce.

Metod pro výpočet těchto koeficientů je několik a jejich výčet a příklad použití můžete nalézt ve zdrojích. V naší aplikaci jsem využil metody využívající průměrování sledovaných hodnot.

$$b_{yx} = \frac{\sum_{i=1}^n x_i y_i - n \bar{x} \bar{y}}{\sum_{i=1}^n x_i^2 - n \bar{x}^2} \quad a_{yx} = \bar{y} - b_{yx} \cdot \bar{x}$$

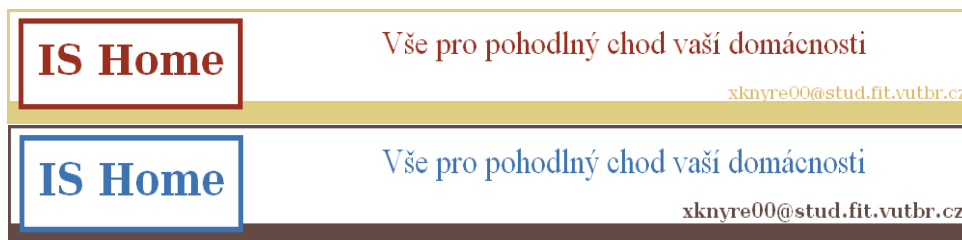
$\bar{x}$  a  $\bar{y}$  jsou průměrné hodnoty zkoumané množiny.[11,12]

## 4.4 Grafické prvky

Při vytváření systému byly použity grafické prvky třetí strany, ale také vlastní bitmapy. Snažil jsem se, aby vzhled aplikace byl co nejvíce "domácký" a pohodlný pro práci uživatele. Zde jsou některé vlastní a převzaté obrázky.



Obrázky znázorňující moduly [13]



Vlastní bannery.

## 5 Testování

Program jsem testoval už v průběhu implementace. Odhalilo to nejen některé funkční nesrovnalosti, ale také chyby v návrhu. Mimo jiné jsem musel vydělit pro tabulku "energy\_types" pomocnou tabulku "units", která definuje jednotky pro jednotlivé energie.

Pro testování funkčnosti systému jsem se snažil namodelovat výdaje průměrné domácnosti. Průměrná data jsem čerpal z Českého statistického úřadu [14] a také z výdajů na energie v mé vlastní rodině.

Dalším krokem bylo testování predikce výdajů. Zde jsem potřeboval historické údaje o počasí na několik let zpátky. Využil jsem přitom internetový portál Wunderground [15], který poskytuje historii stavů počasí pro většinu měst po celém světě.

V průběhu celého vývoje jsem také testoval správnou funkčnost aplikace v nejrozšířenějších prohlížečích. Kromě drobných zobrazovacích rozdílů je aplikace funkční v těchto prohlížečích: Mozilla Firefox, Internet Explorer 7+, Google Chrome a Opera.

## 6 Další možný vývoj

Popsaný projekt popisuje vytváření pouze jednoho modulu Informačního systému pro domácnost. Díky použití Zend Frameworku a modulárnímu přístupu bude jednoduché navázat na stávající práci. Pro budoucnost je možné vyvíjet další moduly a také vylepšovat již existující modul Energie.

Bohužel kvůli nedostatku času a delšímu studování problematiky, nebylo možné realizovat všechny myšlenky na funkčnost systému. Proto výsledný program neobsahuje takové množství funkcí, které se plánovalo při zahájení práce.

Programu by určitě prospěl lepší systém přehledů dat. Dále by šlo zvýšit počet přehledů a vzájemných závislostí mezi daty. Také by se dala vylepšit samotná prezentace dat uživateli použitím vhodnějších kaskádových stylů a modernějších grafických prvků.

Důležitým vylepšením by také bylo ukládání do databáze aktuálních informací o počasí. Současná situace umožňuje pouze ruční nahrávání dat. Tento krok by se dal realizovat automatickým stahováním informací ze serverů, které nabízejí aktuální stavy počasí.

Posledním z možných vylepšení by bylo větší využití Zend Frameworku. Jedná se ale o velice rozsáhlý systém a naučit se jej využívat naplno by zabralo spoustu času.

## 7 Závěr

Bakalářská práce se zabývá návrhem a vytvořením modulu Energie pro Informační systém pro domácnost. V úvodu práce je přehled pojmů a základních znalostí o Informačních systémech pro domácnost. Dále následuje návrh a popis implementace modulu Energie, kde jsou popsány jednotlivé složky tohoto modulu a postupy jejich vytváření.

Práci jsem pojal jako návod na vytvoření tohoto modulu a soustředil jsem se hlavně na podrobný návrh. Tento návrh lze použít při vývoji stejných nebo podobných aplikací. Pro co nejlepší namodelování této problematiky jsem v průběhu vypracovávání práce shromažďoval informace o výdajích domácnosti za energie a důležitých principech informačních systémů.

Výsledný systém zvládá základní funkčnost modulu Energie. U některých rozšířených možností je implementace pouze naznačená a je otevřena dalšímu vývoji.

Práce na tomto projektu mi umožnila vyzkoušet si, vývoj složitější webové aplikace a hlavně jsem se seznámil s možnostmi Zend Frameworku, který i na dále hodlám studovat. Projekt prohloubil moje znalosti ve tvorbě a návrhu informačních systému. Zjistil jsem, na které procesy je potřeba se zaměřit, a jaké problémy při tomto vývoji mohou vzniknout.

Tento projekt mne inspiroval do té míry, že ve vývoji hodlám pokračovat a výsledný projekt sám používat pro jeho široké využití. Doufám také, že nebudu jediným budoucím uživatelem a tento systém budu moci rozšířit mezi širší veřejnost.

Musím konstatovat, že celý projekt mi přinesl hlubší porozumění problematice již zmíněné, a doufám, že pro tyto zkušenosti najdu co nejlepší využití v budoucnu.



# Literatura

- [1] ČEZ a. s., Postup při výpočtu celkové roční platby za elektřinu. Dokument dostupný na URL:  
<http://www.cez.cz/cs/produkty-a-sluzby/obcane-a-domacnosti/elektricka-energie/basic/d-standard.html>
- [2] RWE a. s., Ceny zemního plynu pro kategorii Domácnost. Dokument dostupný na URL:  
<http://www.rwe.cz/miranda2/export/sites/www.rwe.cz/cs/sd/galerie-download/ceniky/Ceniky-ZP-2009/jmp/JMP-04-prehled-cen-2q.pdf>
- [3] Aleš Müller, Energie pod palcem - Základní informace. Dokument dostupný na URL:  
[http://www.santy.cz/energie/info\\_zakladni.php](http://www.santy.cz/energie/info_zakladni.php)
- [4] pepinator.tym.cz, Elektřina - výpočet platby za elektřinu. Dokument dostupný na URL:  
<http://pepinator.tym.cz/download/elektrina.php>
- [5] Michal Adámek, Domáci hospodářství. Dokument dostupný na URL:  
<http://www.sksoft.cz/domaci-hospodarstvi/>
- [6] Entity-relationship model. Dokument dostupný na URL:  
[http://cs.wikipedia.org/wiki/Entity-relationship\\_model](http://cs.wikipedia.org/wiki/Entity-relationship_model)
- [7] Václav Stoupa, Přehled a vývoj PHP frameworků. Dokument dostupný na URL:  
<http://www.root.cz/clanky/prehled-a-vyvoj-php-frameworku/>
- [8] Dušan Janovský, Jak psát web. Dokument dostupný na URL:  
<http://www.jakpsatweb.cz/>
- [9] Zend Technologies Ltd., Programmer's Reference Guide. Dokument dostupný na URL:  
<http://framework.zend.com/manual/en/>
- [10] Coolest DHTML Calendar. Dokument dostupný na URL:  
<http://www.dynarch.com/projects/calendar/>
- [11] Jednoduchá lineární regrese a korelace. Dokument dostupný na URL:  
[old.mendelu.cz/~adamec/regkor.doc](http://old.mendelu.cz/~adamec/regkor.doc)
- [12] Lineární regrese. Dokument dostupný na URL:  
[http://cs.wikipedia.org/wiki/Lineární\\_regrese](http://cs.wikipedia.org/wiki/Lineární_regrese)
- [13] [www.troyboydesign.com](http://www.troyboydesign.com)
- [14] Český statistický úřad, Složení domácností, přehled příjmů a vydání. URL:  
<http://czso.cz/csu/2008edicniplan.nsf/p/3001-08>
- [15] <http://www.wunderground.com/>

# Seznam příloh

Příloha 1. Ukázka vzhledu aplikace

Příloha 2. CD se zdrojovými kódy aplikace, testovacími daty a elektronickou verzí této práce

# Příloha 1

IS Home

Vše pro pohodlný chod vaší domácnosti

IS Home:Énergie

Nastavení

Help

xknyre00@stud.fit.vutbr.cz



ENERGIE

Výdaje za energii

Paušály energií

Výdaje za paušály energií

Jednotky

Cena za jednotku

MĚŘIDLA

SPOTŘEBIČE

INTERVALY

PŘEHLEDY

PREDIKCE

## Přehled typu energi

Přidat nový typ energie

Název	Jednotka	Popis	
Elektrická	MWh	Hlavní jistič	
Plyn	metr3		
Voda	litry		

Přidat nový typ energie